	Type	L #	Hits	Search Text	DBs	Time Stamp
1	BRS	L1	4421	(simulat$4 or model$4) near5 business	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB; USOCR	2003/07/22 16:02
2	BRS	L2	7773	(simulat$4 or model$4) near5 company	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB; USOCR	2003/07/22 16:02
3	BRS	L3	10998	(simulat$4 or model$4) near5 corporat$4	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB; USOCR	2003/07/22 16:02
4	BRS	L4	930	(simulat$4 or model$4) near5 enterpris$4	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB; USOCR	2003/07/22 16:02
5	BRS	L5	658	(simulat$4 or model$4) near5 commerce	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB; USOCR	2003/07/22 16:03
6	BRS	L6	1472	(1 or 2 or 3 or 4 or 5) near5 (framework or architecture or process or procedure or structure or organiz$6)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB; USOCR	2003/07/22 16:04
7	BRS	L7	229181	(information or technology) near5 (framework or architecture or process or procedure or structure or organiz$6)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB; USOCR	2003/07/22 16:04
8	BRS	L8	3433	(business or company or corporat$4 or enterpris$4 or commerce) near5 7	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB; USOCR	2003/07/22 16:04

	Type	L #	Hits	Search Text	DBs	Time Stamp
9	BRS	L9	119	6 same 8 Scanned Ti, Ab, Kwic all	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB; USOCR	2003/07/22 16:06

	Document ID	Issue Date	Inventor	Current OR	Current XRef	Pages
1	US 20020049573 A	20020425	ABU EL ATA, N A			13
2	WO 200219148 A	20030528	ABU EL ATA, N A			27
3	JP 09319796 A	19971212	KITANI, KAZUNORI et al.			9
4	US 6560569 B1	20030506	Abu El Ata, Nabil A.	703/2	703/21; 703/22; 705/7; 705/8; 709/223; 709/224; 709/225	25
5	US 6442557 B1	20020827	Buteau, Brandon L. et al.	707/102	705/7; 707/3	25
6	US 6311144 B1	20011030	Abu El Ata, Nabil A.	703/2	703/13; 703/21; 703/6; 705/1; 705/35; 705/7; 709/220; 709/221; 709/222; 709/223; 709/226	23

L9 results

	Document ID	Issue Date	Inventor	Current OR	Current XRef	Pages
7	US 6236977 B1	20010522	Verba, Stephen M. et al.	705/10	705/14; 705/26; 705/27; 705/37	33
8	US 6233537 B1	20010515	Gryphon, Robert L. et al.	703/1	703/6; 705/7; 707/100	18
9	US 6134706 A	20001017	Carey, James et al.	717/102	345/853; 705/1; 707/9; 717/104; 717/108	11
10	US 5406477 A	19950411	Harhen, John	703/6	705/7; 706/10; 706/46; 706/925; 709/100	63
11	US 5233513 A	19930803	Doyle, William P.	705/7	705/1	272
12	US 20020049573 A1	20020425	El Ata, Nabil A. Abu	703/2		13

L 9 results

US-PAT-NO: 6442557
DOCUMENT-IDENTIFIER: US 6442557 B1
TITLE: Evaluation of enterprise architecture model including relational database
DATE-ISSUED: August 27, 2002
INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Buteau; Brandon L.	Hollis	NH	N/A	N/A
Moulton; Christine S.	Hollis	NH	N/A	N/A

US-CL-CURRENT: 707/102, 705/7 , 707/3

ABSTRACT: A memory is provided for storing data for access by a database program being executed on a computer system for evaluating an enterprise architecture. A data structure is stored in the memory with the data structure including information resident in a database used by the database program. The data structure includes a work flow model, an information model; and a technology model. Each model includes a plurality of entities linking the models together. The computer system executes the database program for evaluating linkages between entities and how architectural changes to the enterprise affect the enterprise architecture by accessing the memory storing the data structure, and generates a result indicative of the linkages between entities.
30 Claims, 10 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 10

---------- KWIC ---------

Brief Summary Text - BSTX (9): To develop and maintain such a model, a wide variety of information about the current enterprise architecture must be collected and analyzed. The TAFIM also provides detailed guidance about what information should be collected, and it supplies detailed examples of data collection forms for this purpose. Unfortunately, the linkage between these forms and the enterprise architecture model is not fully defined, and there are many implicit relationships between the forms that do not appear in the model at all. Therefore, a more detailed data model is required to implement an effective database for storing TAFIM-compatible enterprise architecture data.

Detailed Description Text - DETX (14): Referring to FIG. 4, depicted is a data structure of a work flow model 200 according to the present invention. Examples of information included in the work flow model 200 include descriptions of the organizations, processes and locations and the relationships between them. In FIG. 5, depicted is a data structure of an information model 500. Examples of information included in an architecture include the types of information that the enterprise uses, the formats of the information, the types of repositories that the information is stored in, and so forth. In FIG. 6, depicted is a data structure of a technology model or architecture 600. Common examples of information included in a technology architecture include the hardware and software that support the enterprise. FIG. 7 combines the data structures of FIGS. 4-6 into an overall enterprise architecture model 900.

Detailed Description Text - DETX (18): The work flow model 200 is depicted in FIG. 4 and includes a plurality of entities. An information access entity 210 includes information recording the relationship between a particular process role information access type, and information repository, as well as other access-specific attributes (e.g., frequency of this type of access). To perform different roles in support of an enterprise process, people working in the enterprise need to be able to access information in repositories in different ways. For example, a timesheet entry clerk performing a data entry role needs to "create" access to a labor hours data repository on a weekly basis. A primary key uniquely identifies each instance of this entity by combining three keys inherited from the process role entity 220, information access type 510 (FIG. 5), and information repository entity 520 (FIG. 5). Each information access entity 210 must be associated with exactly one instance of each of these entities 220, 510, 520. The attributes of the information access entity are its frequency (number) and an explanatory annotation (text). This entity is necessary to represent the relationship between process roles and information repositories; and it stands as a common link between the work flow model 200 and the information model 500 of the enterprise architecture model 900 (FIG. 7).

Detailed Description Text - DETX (20): An implementation use entity 230 records a relationship between a particular process role and an implementation. People in an organization perform process roles, and rely upon implementations of one or more automated services to perform these process roles. A primary key uniquely identifies each instance of this entity 230 by a combination of two keys inherited from the process role entity 220 and implementation entity 610 (FIG. 6). The only attribute of the implementation use entity 230 is an explanatory annotation (text). Each implementation use instance must be associated with exactly one instance of each of the process role entity 220 and the implementation entity 610. The implementation use entity 230 stands as a common link between the work flow model 200 and the technology model 600 of the enterprise architecture model 900 (FIG. 7).

Detailed Description Text - DETX (33): The processes entity is depicted at 360. A process is an enterprise activity defined by what the enterprise does during that activity, the information it uses, the organizations and locations it involves, and the result it produces. At a high level of description it is more conventional to refer to these activities as function rather than as processes. Functions are the principal strategic activities of an enterprise. They are often specified in a hierarchical structure; the enterprise as a whole performs some single function or a limited set of major functions, which can each be broken down into subfunctions, and so on. A function simply describes what an enterprise does, independently of the who, where, and how of those activities. For example, one function of an enterprise may be to market its products/services to its customers. This may involve subfunctions of customer analysis, direct customer contact, proposal generation, advertising, and so on. At some point these functions cannot be further distinguished without reference to specific; enterprise information, organizations, and locations. At this level of description (functions in some specific context within the enterprise), the activities are processes. A pure function definition specifies what an enterprise does independently of any other part of the model, while processes are intertwined with information, locations, and organizations. The process entity can therefore record these activity descriptions at any desired level of abstraction; pure "function" processes will have relationships only with other process entities, while lower-level

processes will have relationships with at least some non-process entities. Each instance of the process entity 360 is uniquely identified by an arbitrary integer. The attributes of a process are a text identifier, a textual description of its activity, a textual description of its method, and a textual description of its result (which should be a tangible product or measurable service; information results should be identified through an information relationship). The process entity 360 is used by a process relationship entity 380 with each process relationship links two processes. The process entity 360 is used by an information relationship entity 390, organization role entity 300, process role entity 220; and process location entity 330. This entity is an essential part of the enterprise work flow architecture.

Detailed Description Text - DETX (36): The information relationships entity 390 describes how the different types of information used by an enterprise are produced (output) or consumed (input) by many different process. The information relationship entity 390 records the relationship between a particular information type and process, including attributes that specify whether that relationship includes input, output, or both properties. Each instance of the information relationships entity 390 is uniquely identified by a combination of two keys inherited from the information type entity 540 (FIG. 5) and process entity 360. The attributes of the information relationship entity 390 are an input predicate (binary), an output predicate (binary), and an explanatory annotation (text). Each information relationship entity 390 must be associated with exactly one instance of each of the information type entity 540 (FIG. 5) and process entity 360. The information relationships entity 390 is necessary to represent the relationship between information types and processes. If more types of information relationships are desired (besides input and output), then the two binary attributes of this entity could be replaced with a single text attribute or even by a separate entity (i.e., an information relationship type entity). The information relationship entity 390 stands as a common link between the work flow model and the technology model of the enterprise architecture.

Detailed Description Text - DETX (43): A technology acquisitions entity 460 is depicted in FIG. 4. Multiple organizations within an enterprise may acquire a variety of technology items and pay different amounts for the same items over a variety of time periods. The technology acquisition entity records this relationship between the organization entity 290 and technology acquisition item entity 650 (FIG. 6), along with the useful date and cost information about each acquisition. Each instance of this entity is uniquely identified by a primary key using an arbitrary integer. The attributes of the technology acquisition entity 460 are an acquisition date, an expiration date (optional), a binary flag identifying whether the acquisition is a purchase (a non-purchase indicating a lease or license), a binary flag indicating whether the acquired technology can be used after its expiration date, a one time cost (currency), an annual cost (currency), and an explanatory annotation (text). The technology acquisition entity 460 uses the organization entity 290; each technology acquisition must be associated with exactly one acquiring organization. In addition, each technology acquisition may be associated with one supplying organization. Finally, this entity is used by the technology acquisition item entity 650, which associates technology acquisitions with one or more specific technology items. Although not an essential part of an enterprise architecture model, this cost and date information is very useful to information technology decision makers, so it makes, sense to make a place for this kind of information in the enterprise

architecture database.

Detailed Description Text - DETX (63): A systems entity 730 is depicted in FIG. 6. Baselining the specific information technology of an enterprise often begins by identifying systems: collections of hardware and software that provide interrelated automated services. Typically, systems are identified by the enterprise itself on the basis of what technology is used together to support some process. From the perspective of this enterprise architecture model, however, a system is defined to be a collection of implementations of some services; each implementation itself represents one or more technology items. Each instance of this entity is uniquely identified by a primary key using an arbitrary integer. The attributes of a system are its acronym, name, a textual description, and a textual comment. The system entity 730 is used by the system component entity 680, which associates a system with one or more implementations and vice versa. The systems entity 730 is an important part of the enterprise technology architecture. From the perspective of the essential logical design of an enterprise architecture, the system entity may appear to be more useful than it actually is. Automated systems have definitions or boundaries that are often determined by whomever pays to build or maintain the technology, rather than by any consistent logic of enterprise-wide technology use. (This is one reason for creating the implementation entity.) In spite of this problem, systems are probably the technology entities most recognizable to enterprise decision makers and IT managers, and most easily available baseline information is probably collected with respect to systems, so it makes sense to incorporate them in this database.

Claims Text - CLTX (27): 27. A computer-readable medium having a data structure representing an enterprise architecture stored thereon for access by a data processing system to evaluate the enterprise architecture, the data structure comprising: a work flow architecture model including a plurality of entities; a technology architecture model including a plurality of entities; and an information architecture model including a plurality of entities; wherein the work flow architecture model, information architecture model and technology architecture model form an enterprise architecture; wherein the computer system executes the database program for accessing the memory store and evaluating linkages within and among the work flow architecture model, information architecture model, and technology architecture models; wherein, in order to make decisions about business restructuring, internal technology investment, or enterprise-level system architectures, a user analyzes dependencies between the structure and function of an enterprise and the information technology relied upon by the enterprise to determine the impact of information technology changes upon enterprise structure and function.

US-PAT-NO: 6560569
DOCUMENT-IDENTIFIER: US 6560569 B1
TITLE: Method and apparatus for designing and analyzing information systems using multi-layer mathematical models
DATE-ISSUED: May 6, 2003
INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Abu El Ata; Nabil A.	Omaha	NE	68154	N/A

US-CL-CURRENT: 703/2, 703/21 , 703/22 , 705/7 , 705/8 , 709/223 , 709/224 , 709/225

ABSTRACT: An information design system uses an input module, a construction module, a performance metrics module, and an output module to create and output several models of a proposed information design system. The input module receives descriptive input which is validated and transformed into quantitative input. The construction module uses the quantitative input and information from a library of hardware and software component models to create and calibrate one or more models. The performance metrics module calculates performance metrics for the models, which then can be compared based on these metrics. A preferred information system design can then be selected from the models.
33 Claims, 9 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 9

---------- KWIC ---------

Detailed Description Text - DETX (25): The business management domain is represented in the modeling process through descriptive information for both the business process and the business function layers. These two layers represent the impact of the organization's activities on the performance of its information system.